(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
5 April 2001 (05.04.2001)

PCT

(10) International Publication Number
WO 01/23981 A1

(51) International Patent Classification⁷: G06F 1/00, G06T 1/00

(21) International Application Number: PCT/GB00/03712

(22) International Filing Date:
28 September 2000 (28.09.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
9922904.9 28 September 1999 (28.09.1999) GB

(71) Applicant *(for all designated States except US)*: SIGNUM TECHNOLOGIES LIMITED [GB/GB]; 6 Thorney Leys Business Park, Witney, Oxford OX8 7GE (GB).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: HILTON, David [GB/GB]; 12 Harveys Lane, Winchcombe GL54 5QT (GB).

(74) Agent: ORIGIN LIMITED; 52 Muswell Hill Road, London N10 3JR (GB).

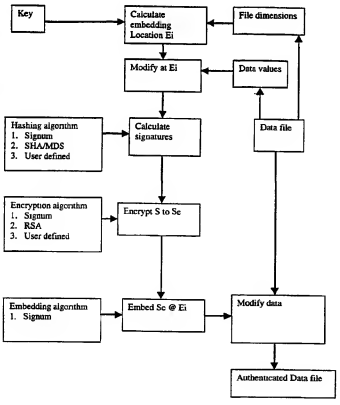
(81) Designated States *(national)*: AU, JP, US.

(84) Designated States *(regional)*: European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published:
— *With international search report.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: AUTHENTIFICATION OF DIGITAL DATA WORKS USING SIGNATURES AND WATERMARKS



(57) Abstract: A method of authenticating a digital data work, to enable its integrity or authenticity to be established, is taught. The method envisages calculating a unique signature for the unaltered work and embedding that signature in some manner back into the unaltered work in a way that actually changes the work itself.

WO 01/23981 A1

AUTHENTICATION OF DIGITAL DATA WORKS USING SIGNATURES AND WATERMARKS

Field of the invention

This invention relates to a method of authenticating digital data works, particularly to enable any unauthorised alterations to that work to be rendered readily detectable.

Description of the Prior Art

Systems based upon digital data are becoming universal and indispensable; digital data passing between computers; digital telecommunications; digital audio; digital cameras; and the convergence of many of these individual components driven by the internet, set the prior art context to this invention.

There are many applications for a technique that can enable any unauthorised alterations to digital images or digital audio to be rendered readily detectable. For example, it is today very easy to tamper with a digital photograph using image manipulation software, rendering the authenticity of any digital photograph questionable. This has serious implications for the use of digital photographic evidence in criminal litigation, for example. It may therefore be advantageous to be able to assert that the integrity of any given digital photograph can be assured. Similarly, it is becoming common to archive documents, including legal contracts and financial instruments, by imaging and storage on non-erasable digital media, such as WORM. There is a pressing need to ensure that those digital records are tamper evident. There are similar issues pertaining to digital audio and video. For example, where digitally recorded speech is to be used in evidence, typically to confirm the existence and terms of an oral agreement, then validation of the integrity of the recording is particularly helpful.

There are various established approaches to ensuring data integrity in the telecommunications and digital audio fields. For example, the use of error correction techniques relying upon check-sums. However, these techniques are designed to ensure that a digital signal generated by a device, for example, a CD player, is an accurate transmission or reproduction from a source, for example the data stored within the CD. That is different

from being able to detect if, without access to the original CD, any duplicate made of the CD is a completely accurate reproduction of the original CD. The present invention is directed to solving this latter problem, i.e. whether the integrity of a digital data work has been compromised. Hence, the present invention is not directed to manipulating small units of digital data to enable that data to carry information inherent to the proper comprehension of the digital data itself but instead to enabling any unauthorised modification or alteration to a digital work to be readily detectable.

Digital signatures are one technique widely used to confirm the integrity of digital information, particularly to verify the integrity of digital information which has been transmitted from one location to another. Currently, it is possible to include a simple signature in the header of the data file of a digital data work. The header might typically comprise a checksum derived from the contents of the data file so that any alteration of the contents inevitably leads to a mismatch with the checksum; the mismatch can readily be detected, enabling the alteration to be detected. However, it can be relatively easy to strip out the header checksum entirely, in which case one could not establish the integrity of the data.

Hence, the conventional, signature based approach to checking the integrity of a digital file is as follows: the person who wishes to authenticate a digital file calculates a signature by applying an appropriate algorithm to the data values of the file. This signature is then either appended at the end of the original file or written into a separate file. The signature is transmitted along with the file to anyone who wishes to use the file and have corroborating evidence of its authenticity. The verification part of the process is carried out by the receiver of the file. The receiver must have knowledge of the algorithm used to calculate the signature together with any key that governs that algorithm. Using these tools the signature is calculated for the received file. The value of the signature that was transmitted is then compared with the signature that has just been calculated. Equality of values indicates that, to whatever probability the method entails, the transmitted file is indeed an identical copy of the original file.

A further possibility as described by EP-A-0402210 is to write the signature into a secure hardware device. EP-A-0402210 describes verifying the integrity of a message by looking at the unique signature which can be derived from a part of the message and comparing it to the signature derived from the original of the message, stored in a secure location.

Reference should also be made to EP-A-96920963.4, which discloses the approach of enabling the integrity or authenticity of a digital data work to be established by changing the digital data work according to a particular algorithm so that some or all of its constituent parts possess a measurable characteristic; that measurable characteristic is changed if any alterations to the digital data work are subsequently carried out. That change can then be detected using a detection process.

Reference may also be made to US 5613004 which discloses embedding an invisible watermarked message into an image; a hash of that message is also embedded, enabling one to detect whether the message has been altered. The signature is therefore not of the work as such but the message embedded into it.

Statement of the invention

In accordance with the present invention, a method of authenticating a digital data work, comprises the steps of:

calculating a signature for an original, unaltered version of the digital data work;

modifying some or all of the original, unaltered version of the digital data work in dependence on the signature, to generate a modified work, such that an unauthorised alteration to the modified work measurably changes the modified work.

Hence, the essence of the invention is to enable the integrity or authenticity of a digital data work to be established by calculating a unique signature or hash for the unaltered work and embedding that signature in some manner back into the unaltered work in a way that actually changes the work itself. That contrasts with the conventional approach of merely appending a signature as a header or some other adjunct to the data work. It contrasts also with embedding a message and a hash of that message to guarantee the integrity of the message. Although the present invention is based on the concept taught in EP-A-96920963.4 of actually altering the digital data work, this invention goes beyond that since EP-A-96920963.4 does not disclose the concept of altering the digital data work by embedding in it a unique signature derived from the original data work itself.

The signature is embedded into the digital work itself such that any usage of the original file will typically be unaffected by the signature calculation; indeed it is unlikely that any user will be aware of the presence of the embedded information.

The approach of the present invention has two important consequences. The first is that an additional level of security is possible over conventional techniques because the location and manner in which the signature is embedded into the original data work allows for a further form of encryption. The second consequence is that the parts of the data into which the signature is encoded have to be handled with specially devised algorithms which will allow them to contribute to the overall signature in such a way that the contributions will not be affected by the signature encoding process applied to the digital image.

A preferred method applies equally well to image files compressed in the JPEG format and to audio files compressed in the MP3 format, although in each of these cases the encoding of the signature into the data requires additional procedures to avoid it being damaging to the quality of the data work.

In one possible implementation of this method, a signature is calculated using only part of a digital data file, and the calculated signature is embedded into this selected part. Thus, for instance, a document may be scanned into digital form and then areas of the document that are particularly significant (e.g. hand-written signatures on a cheque or contract) may be chosen. For each of these significant areas a signature may be calculated and embedded into the data. Thus an alteration to one of these selected areas could be detected whilst the remainder of the document could be certified as unchanged.

A further variant occurs in the case of JPEG files. In this case it may be that it is important in some context to have a signature that is based on the whole file including the header. However, when the signature is embedded, it has to be embedded in a particular part of the data (see below) because some parts of the data carry critical information in a form such that any alteration renders the information meaningless.

Likewise with MPEG files: a signature may be based purely on the data, or on the whole file including the header, but the embedding has to be carried out on the data values alone.

The nature of the signature depends upon the requirements in the context under consideration. Algorithms for calculating signatures, known as 'hashing functions', have been studied at length and there are secure versions in common use. One such algorithm, known as SHA-1, invented in 1994, has a high level of security. A signature of this form is really a highly condensed digest of the digital data; it may be that a megabyte of information is mapped onto 64 bits of signature. Clearly there are many possible images that map onto any one signature, but with a good hashing algorithm it is virtually impossible to find another image or audio file that has the same signature as that calculated from the original data. In the method envisaged for one embodiment of the present invention, an algorithm such as SHA1 may be ideal for circumstances in which any alteration to the original data, however small, is to be calculated (the "Type A" method of authentication as described in this specification).

The present invention also deals with the possibility that a file may be altered to a small degree but still be acceptable ("Type B"). Two common situations occur where Type B processes might be the useful. The first is where files are compressed but only to the extent that image or audio reproduction is scarcely perceptibly altered. The second is where image files are realised in hard copy and hence have to go through a screening or equivalent process and yet are sufficiently high quality to be acceptable as authentic versions of the original. In these cases this specification describes how a signature might be used to provide a guarantee of integrity and a measure of the extent to which changes have occurred.

In this latter, Type B case, the signature needs a property which is specifically avoided in algorithms of the SHA-1 type. That is the property that a small change in the image will produce a small change in the signature. The reason for this is that using SHA-1 an operation such as compression might give a totally different value of the signature from that corresponding to the original. In the Type B method described in this specification, the variation in signature would be usable as a measure of the degradation that has occurred.

This latter type of signature has the drawback that it is more possible to produce two images with the same signature, but by careful choice of function this risk can be made acceptably small.

These 'approximate' signatures for Type B applications will typically use sets of orthogonal functions to develop an image description. These functions will vary according to a key so that it will not be possible to develop a general method for making known modifications to the signature. Alternatively, the signature may be calculated by taking a random selection of pixels from the data, selecting the pixels in such a way that anyone attempting to alter the file would have difficulty in avoiding all of the sampling points.

In one embodiment, the original digital data work is modified not simply by a signature, but also by an externally generated code. That code may form part of a copyright management system.

In another aspect of the invention, there is provided a method of detecting any alteration of a digital data work, to which the authentication method of the present invention has been applied, comprising the steps of:

reading a signature embedded into the modified work;

calculating a signature for the modified work;

comparing the embedded signature read from the work with the calculated signature and determining that the modified work has been altered if the embedded signature does not match or otherwise correspond to the calculated signature.

This enables the authenticity of the digital data work to be confirmed or rejected.

In further aspects of the invention, there is provided adigital data work to which the authentication method encompassed by the invention has been applied; a computer program operable to authenticate a digital data work using such an authentication method, a computer program operable to detect any alteration to a digital data work, to which such an authentication method has been applied, and, finally, digital media pre-recorded with such a computer program.

Brief Description of the Drawings

The invention will be described with reference to the accompanying drawings in which:

Figure 1 is a flow diagram of the essential steps performed in one embodiment of the present invention.

Detailed Description

Signature Embedding

In a preferred embodiment of the present invention, the embedding of the signature into original data is performed in such a way that (i) the file formats of the original data are unchanged and (ii) any changes in the file data will be sufficiently small as to be imperceptible when the files are realised as images or sound clips. The nature of this embedding depends on whether we are considering Type A or Type B: Type A authentication (as noted above) is designed to detect any modifications however small, whereas with Type B authentication, the embedded signature may have to survive minor, legitimate manipulation.

Referring now to Type A methods, some spatial domain (as opposed to frequency domain) modification of the data work must take place in the actual process of authentication, but the number of sites at which data is modified to embed a signature must be kept to a minimum. In order to give added security to the process these sites are selected by a process which is dependent on a crypto key unique to a user. Obviously, many possible means of selection are available. Random number generators offer a simple method. An alternative method used in previous fingerprinting applications (such as EP-A-96904936.0) is the selection by the use of permutations.

The embedding of a signature for JPEG files requires a greater degree of differentiation of data. In the JPEG format, part of the information is encoded in the form of 'Huffman' code, a variable length coding in which values that frequently occur are encoded with shorter

length codes than those of rarer occurrence. The problem with this sort of code is that it cannot be modified to contain information without rendering the code unreadable. For this reason the embedding of the signature in a JPEG file has to be carried out in the part of the file which simply gives the magnitude of the data values after the discrete cosine transformation (DCT). These changes will affect the quality of the information to a minor degree. The perceptibility of these changes is minimised by selecting from areas with a large number of non-zero coefficients in the DCT. The actual selection may be governed by the encoding key, adding a further level of security. These areas correspond to parts of images where there is a lot of change present rather than to smooth areas and are areas where alterations may be carried out with little effect on image quality. For JPEG data, a signature could be calculated for a complete file, including the header, of for the data part alone, or for some subset of the data. In each of these cases, however, the restriction as to the mode of embedding that is described above must apply. As for non-compressed files the image is divided into subsections. In the case of JPEG files subdivision is carried out by insertion of "Restart Markers." A hash value is calculated using all of the data that describes a given subsection without regard to the interpretation of that data. In one embodiment the hash value includes the Huffman and Quantisation tables which are mandatory parts of the header .

The situation with MPEG files involves another level of complexity. In this case the data following the header is stored frame by frame, but in order to achieve compression there exist certain reference frames which are placed at regular intervals and enable intermediate frames to be encoded in a more economic manner. One method of handling the signature is by calculating the signature value on all frames but restricting the embedding to the reference frames in a manner similar to that employed for JPEG files.

MP3files, (defined in ISO/IEC 11172-3) like JPEG files, have compressed format with variable length code. Again like JPEG files it is essential to modify values in such a way that whilst there is a mild degradation of audio quality there is no risk to the whole structure and interpretation of the files.

As in previous cases an MP3 file is subdivided into subsections, a hash value calculated for each subsection and then embedded back into that subsection.

In the case of MP3, data is divided into fixed length subsections as part of the compression algorithm. This is because the audio data is analysed into a range of frequencies so that certain psycho audio effects may be taken into account to allow removal of data which will produce sounds that are masked by louder sounds on neighbouring frequencies. The data is then expressed as amplitudes of a range of frequencies, complicated by the inclusion of scaling factors.

Two embodiments are cited here. In the first the hash values are embedded in the "scale factors" which are described in the MP3 specification. There are several scale factors in each frame of MP3 data and so suitable factors can be chosen for amendment, chosen in such a way as to produce minimal perceptual impact. In the second the hash value is embedded by modifying the "preflag" (see MP3 spec.), a quantity which occurs more sparsely than scale factors, but nonetheless a quantity which can be modified without damaging degradation to the audio quality.

With Type B authentication, the embedded signature may have to survive minor manipulation, as mentioned previously. In this case the signature will be embedded in the form of an invisible fingerprint or watermark (see EP-A-96904936.0). The essence of the watermarking is that a large number of data values are affected but only modified by a very small amount. The result is that if such a file were to be compressed, for instance, the embedded signature could still be read from the watermark and the signature itself, calculated from the data in the file, would have an approximately equal value.

A fundamental feature of the method of authentication described herein is that alterations may be detected if data is cropped, or copied and pasted from authenticated files.

Methods of authentication have been in use in which, as described in this patent, a hash value is obtained to describe a set of data, and the hash value is dependent upon a key which should be unique to the user. The hash value is then appended to the file or, as in this proposal, embedded back into the file. In some cases the hash value describes a complete file and hence although an alteration may be detected in the file there is no means to determine the location in which the alteration has been made.

In some cases, a file may be subdivided into sections and a hash value determined for each section. In these cases it will be possible if an alteration has occurred not only to detect that such an alteration has taken place, but also to locate the alteration within one of the subsections.

Two possible weaknesses occur in the above scenario. The first is that cutting and pasting may not be detected, the second that cropping of files may not be detected. This is best illustrated by examples.

Suppose that authentication is used to protect a set of scanned cheques. The authentication may subdivide the image into convenient subsections, calculate a hash value for each subsection and embed the hash value back into the file. The manner of embedding the hash value may depend upon a key which is unique to the user of the software, and this key will be used to select the sites at which the hash value is embedded. The whole set of cheques may be authenticated with the same key. If one of the areas of subdivision is the section of the cheque which indicates the amount to be transferred, this area will be treated exactly the same on each cheque in the sense that the same hash function will be used and the same embedding points selected to embed the hash value. If now this important area of the cheque is copied from one cheque onto another the presence of the forgery will not be detected because the hash value will be the one that matches the data.

The second problem, that of cropping, is a serious issue for images and audio clips. A forensic image may be totally different in interpretation if some part is cropped, perhaps some bystander being removed from a scene of criminal activity. The meaning of an audio

clip may be reversed by omission of some introductory qualifying phrase. Now again, preceding patents as described would confirm the authenticity of a given file even if there were parts omitted, provided that those parts were complete subdivisions of the original.

There are several embodiments of the present invention which overcome the above objections. In one embodiment the hash value calculated for a given subdivision of an image or audio clip includes a value which indicates the position in the file of that subdivision. For example, if an image is divided into rectangles the distance of each edge of the rectangle from the edges of the original file might be included in the hash value. If then any cropping of the image occurs the distances to the edge would be altered and the hash value falsified.

In another embodiment the hash value or any subsection depends upon values in neighbouring subsections. The number of values can be chosen according to the size of area within which an alteration may be detected, and upon the probability of false values giving rise to the same hash value. In this case if an area were to be cut and pasted the incorrect values for surrounding areas would corrupt the hash value, allowing detection of the counterfeit.

Detail of Type A Authentication: detection of any change, however small

Suppose we have a set D of digital values $\{d_i\}$.

Each user has a key, K, and this key is used to select a subset D_e, small in comparison with D, into which the signature will be embedded.

Thus $D = D_u + D_e$ .....(1)

where D_u is the set of values that will be unchanged.

Thus we might embed 128 bits of signature into a 1 megabyte file.

An algorithm, A, is devised which maps a set of values onto a single value, S, the signature of the data. At its simplest this might simply be a case of adding each pixel value multiplied by the x and y coordinates and then neglecting the overflow if values exceeded 128 bits. Or, for an audio file, multiplying the amplitude value by the position in the file and accumulating values similarly to image files.

We need to calculate the signature using the values in D_e as well as those in the unmodified set D_u. This will prevent alterations to the set D_e going undetected. The problem is that the values of D_e which the authenticator will see may be different from those that the detector sees because they have been modified in the process of embedding the signature. To cope with this it is necessary to use modified values from D_e to calculate the signature, modified in such a way that the signature calculation will be unaffected by changes required for the embedding process (see example below).

In mathematical terms a mapping, M, and a coding algorithm, C, are constructed such that:-

M maps the set of values in D_e onto a new set , $D_{e,m}$.

$$M(D_e) = D_{e,m} \text{.....(2)}$$

A signature for the data set D is calculated . This signature uses the values in the part of the data, D_u, which is unchanged together with the modified values of the embedding set, D_e.

$$S = A\,(D_u + D_{e,m}) \text{ .....(2A)}$$

Once the signature is calculated it needs to be embedded into D_e by a coding algorithm C. A very simple method is to express S as a binary string and embed it by changing values of D_e to even numbers to represent '0' and to odd numbers to represent '1.'

Thus, the value of S is coded into the values D_e by algorithm C,

$$C(S,D_e) = D_{e,c}. \text{ .....(3)}$$

The mapping M must have the required property,

$$M(D_e) = M(C(S,D_{e,m})) \ldots\ldots\ldots\ldots\ldots(4)$$

That is, the contribution to the signature by elements in set D_e must be unaltered by the modification of the values to include the coded signature.

Checking of the integrity of the data consists of :-

(i) calculating D_e from key, K.

(ii) mapping D_e onto $D_{e,m}$

(iii) calculating $S = A(D_e + C(S,D_{e,m}))$ ……..(5)

(iv) deriving the embedded signature in C(S,De,m)

(v) comparing derived signature with the S in (iii).

Security

There are 4 areas in which a form of encryption is used and where security can be added.

1. The method of mapping a key, K, onto a selection algorithm which derives the set D_e for embedding the signature.

2. The selection of hashing algorithm, A, for calculating the signature.

3. The choice of coding algorithm, C, such that the set of values of D_e is mapped onto a new set of values.

Part of the security method is contained in the control of access to the software which adds the signature. It is envisaged that the signature will be added at a small number of secure sites but that the detection program will be widely distributed.

Now for most applications a fairly simple level of encryption will be more than sufficient to deter any but the most sophisticated of attacks. However, if the detector is distributed it will be possible to reverse engineer the algorithms. In any case the detector must contain the algorithms A and C and the algorithm for selecting D_e.

Mapping of the key onto the selection of sites

A key may be of virtually any length according to the level of security required. The signature that is calculated may be converted into a binary string of virtually any length and there is then a requirement that the number of sites chosen to embed the string must be equal to that length.

One method of selection of suitable sites is the permutation method described in EP-A-96904936.0, in the name of the present applicant, the text of which is incorporated by reference into this specification. This method relies on an internal permutation to generate a set of permutations which in turn identify a set of locations in a file. These locations are then used to embed the coded signature. The values at these selected sites are then mapped onto new values, by using an algorithm of the type M above, and it is these modified values that are used in the calculation of the signature.

Calculation of the Signature (algorithm 'A' above)

There are many available hashing algorithms with accepted levels of security. The SHA 1 algorithm, for instance, is used by PGP in its applications. Any such algorithm is acceptable to produce a signature to be embedded. However, since the security is not provided entirely by the hashing it is possible to use a simpler hashing algorithm or one with particular properties, and still have a secure signature.

Encoding of Signature ('C' and 'M' above)

The simplest way to encode the signature, S, is probably to take the binary string and embed it at the selected sites using coding, C, such that a '1' bit corresponds to an odd value and a '0' bit corresponds to an even value. This type of coding ensures that no data value needs to be changed by more than one unit. This fact, together with the fact there is only a small

number of sites which need to be amended, ensure that the modifications are minimal and certainly would not be visible.

The mapping, M, for this particular coding algorithm, can simply be the rule "divide by two giving the result as the nearest integer" provided that the coding of odd and even is carried out by use of subtraction and not addition. This is best illustrated by an example.

If original value is 25 and it is to be rendered even then the adjusted value must be 24 and not 26 on the grounds that 25/2 = 24/2 neglecting the fractional part.

i.e. $M(D_e) = M(C(S,D_{e,m}))$ .....(7)

If $D_e = 25$, $C(D_e) = 24$, $M(C(D_e)) = 12$ .....(8)

$M(D_e) = M(25) = 12$ .....(9)

More sophisticated coding algorithms exist and can provide higher security.

For example, supposing that there is a restriction of the coding algorithm that it shall not alter any value by more than one unit, as above, then the coding algorithm may again divide the data values into pairs but the interpretation of each member of the pair may be varied according to a selected rule. Again, illustrating by an example:

Suppose that each member of the data set has a value in the range 0 to 7. Suppose we had 4 sites that were used to embed the signature 1001.

Suppose the original values at these sites were 4,6,3,5

Using the method described above, to embed 1001 these values would be modified to be odd,even,even,odd respectively.

Thus 4,6,3,5 would be mapped onto 3,6,2,5.

The values used to calculate the signature would be half of each of the above values, i.e. they would take the values 2,3,1,2 and this would be the case both for the original and modified set.

Thus in the above system the data values are mapped onto coding values as below:-

(2 ~ 0 to be read as '2 is mapped onto 0')

0 ~ 0,1~1, 2~ 0,3~1,4~0,5~1,6~0,7~1 ……..(10)

However, we could equally well pair the numbers in a different fashion. Thus, for example,

0~0,1~1,2~1,3~0,4~1,5~0,6~0,7~1 …….(11)

If then we wished to embed 1001 in data values originally 4,6,3,5 we would proceed as follows

The value 4 corresponds to the value 1 in (11) and so need not be changed

The value 6 corresponds to the value 0 in (11) and so need not be changed

The value 3 corresponds to the value 0 in (11) and so need not be changed.

The value 5 corresponds to the value 0 in (11) and so must be changed to 4 which corresponds to 1 in (11). Note M(5) = 2 and M(4) =2 so the value used for calculation of the signature is unchanged.

Generalising the above, if the data values range from 0 to n - 1, these values are grouped in pairs. One member of each pair is made to correspond to the value 1, the other corresponds to the value zero. The choice of which corresponds to 1 and which to zero can be by algorithms based on the key or the signature. The best from the security view is to base the algorithm on the signature. For instance, if the signature were to be expressed as a binary string, whenever a 1 occurred the pair of numbers might be allocated in the order 01, whereas when a 0 occurs the pair might be allocated in the order 10. To add to the security any encryption might be used to map the signature onto an encrypted value.

A higher level of security can be provided by the following method. The signature, S, is calculated as above. This is then encrypted by the private key of an asymmetric encryption algorithm, E, to give new signature, S_e. The RSA method provides a suitable algorithm. S_e is embedded in the data by coding algorithm, C.

The detection software has only the public key. The detection process consists of calculating the signature S by the method above and reading the coded embedded signature, S_e. The public key is then used to decode S_e when it can be compared with the value of S.

Detail of Type B Authentication: detection of changes, ignoring small alterations likely to be imposed by legitimate processes

If a document that is at least partly handwritten (cheque or financial document) is scanned as a greyscale image it could be authenticated with an approximate hash function. Thus if it were to be stored in compressed form it would be possible to check whether there had been significant modification to the image. (If the document were to be entirely produced from the keyboard it is likely that OCR would be used.) If a document as described above were to be printed, the approximate hash function could indicate whether or not there had been significant alterations.

If an image needs to be authenticated at point of delivery and before any possible compression it could be represented by an approximate hash function. This hash function would then only be slightly modified by compression or printing and hence could supply a confidence level for such alterations.

Approximate Algorithm

The approximate algorithm works similarly to the Type A method in that a hash value for the entire image is calculated and then embedded back into the document. In the approximate algorithm, however, the embedding cannot be on a simple scheme of

modifying least significant bits because these bits may well be modified by the sort of small changes the algorithm is required to handle. Instead the embedding is in the form of a very light watermark that modifies very slightly every pixel, being rather like the watermarking for medical images where sensitivity is of the essence.

An alternative to embedding the hash value is to store the value in a database in encrypted form. This would allow for a more detailed image description.

The essential feature of the approximate hashing algorithm is that it corresponds to geometrical features of the image. This means that small changes in the image will produce small changes in the calculated signature, notably in the case of JPEG files and printed files the signature will only undergo small changes. The usual emphasis in hashing algorithms is for small changes in data values to produce totally distinct hashing values. It is this distinctness that supplies the security and makes it difficult to produce a second document with the same hashing value as an existing document.

The approximate algorithm must have its security protected by other means. One method is to apply an asymmetric encryption to the hash value before it is written as a watermark. The decoder then needs only the public key to verify integrity and if the security protocol is such that the private key is only in the hands of trusted parties, the security can be high.

The hashing algorithm requires slightly different qualities according to whether it is to be applied to files that remain wholly in the electronic domain and hence will not suffer any change of orientation or aspect ratio, or to files that are to be interpreted in hard copy and rescanned. These two cases are discussed below.

Electronic Files

The hashing algorithm will in essence be a geometrical image descriptor with a high degree of accuracy. The most straightforward descriptors available are first and second order moments which correspond to the notions of centre of mass and principle axes. A set of further descriptors must be added to limit the image manipulation that can be carried out

without detection. One such set could be supplied by the use of moments derived from orthogonal functions.

A method of protecting the security whilst using a simple moment calculation would be to divide the image into several different sets, where the sets are selected according to a user key, and to find linear moments from each set. If linear moments were to be taken without any other protection it would be fairly easy for a user to make alterations which left the centre of mass of the image unaltered However, if the moments are taken from randomly selected subsets of an image no such possibility exists.

The sets would be selected such that groups of pixels of significant size for the document in question would belong to one set. Thus for documents the sets would consist of groups of pixels of roughly the size of a printed character.

Hard Copy Files

In the case of hard copy files the first and second order moments of the whole image would be used to establish scaling and orientation. All subsequent descriptors would be evaluated using these as a co-ordinate system. Again a set of orthogonal functions could provide further descriptors, or, as above, simple moments for selected subsets could be used.

The appended Flow Diagram (figure 1) summaries the essentials of the above processes.

Claims

1. A method of authenticating a digital data work, to enable an unauthorised alteration to that work to be readily detectable, comprising the steps of:

calculating a signature for an original, unaltered version of the digital data work;

modifying the original, unaltered version of the digital data work in dependence on the signature by embedding into it the signature to generate a modified work, such that the unauthorised alteration to the modified work measurably changes the modified work.

2. The method of claim 1 in which a signature is calculated using only part of a digital data work, and the calculated signature is embedded into this part.

3. The method of Claim 2 in which the digital data work is an image and the part of the image from which the signature is calculated and into which the signature is embedded contains the most significant information.

4. The method of Claim 1 in which a signature is calculated using the whole of a digital data file, and the calculated signature is embedded into this whole, other than portions which carry critical information which should not be altered.

5. The method of Claim 1 in which the signature is used to provide a measure of the extent to which acceptable changes have occurred to the digital data work.

6. The method of Claim 5 in which the signature possesses the property that a small change in the data work will produce a small change in the signature so that a variation in signature is usable as a measure of any degradation that has occurred.

7. The method of Claim 1 in which the signature is calculated by taking a random selection of pixels from an image data work, selecting the pixels in such a way that anyone attempting to alter the file would have difficulty in avoiding all of the sampling points.

8. The method of Claim 1 in which the original digital data work is modified by a signature and an externally generated code.

9. The method of Claim 8 in which the externally generated code may form part of a copyright management system.

10. The method of Claim 1 in which the distribution of the signature within the digital data work is selected by a cryptographic process.

11. The method of Claim 10 in which the cryptographic process is a permutation process.

12. The method of Claim 1 in which the data works are MPEG audio files and the signature value is calculated on all frames but the embedding is restricted to the reference frames.

13. The method of Claim 1 in which the data works are JPEG image files and the embedding of the signature is carried out in the part of the file which gives the magnitude of the data values after the discrete cosine transformation.

14. The method of Claim 1 in which the data work is subject to minor permissible manipulations, in which the signature is embedded in the form of an invisible watermark.

15. The method of Claim 14 in which the minor permissible manipulations affect a large number of data values in the watermark, but only by a very small amount, such that the embedded signature is still derivable from the watermark and the signature, calculated from

the data in the file, would have an approximately equal value to the embedded signature.

16. The method of Claim 1 in which the signature is derived using a hashing algorithm selected so that it is virtually impossible to find several images or audio files that generate the same signature as that calculated from the original digital data work.

17. A method of detecting any alteration of a digital data work, to which the authentication method of claims 1-16 has been applied, comprising the steps of:

reading the signature embedded into the modified work;

calculating a signature for the modified work;

comparing the embedded signature read from the work with the calculated signature and determining that the modified work has been altered if the embedded signature does not match or otherwise correspond to the calculated signature .

18. A method of confirming the integrity of a digital data work, to which the authentication method of claims 1-16 has been applied, comprising the steps of:

(a) reading the signature embedded into the modified work;

(b) calculating a signature for the modified work;

(c) comparing the embedded signature read from the work with the calculated signature and

(d) confirming the integrity of the modified work if the embedded signature matches or otherwise corresponds to the calculated signature.

19. A digital data work to which the authentication method of any of claims 1-16 has been applied.

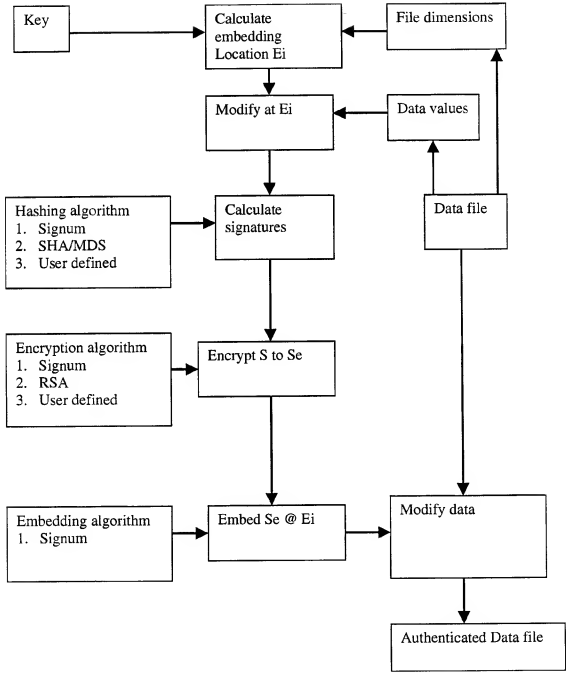
20. A computer program operable to authenticate a digital data work using the method defined in any of Claims1 to 16.

21. A computer program operable to detect any alteration to a digital data work, to which the authentication method of claims 1-16 has been applied, using the method defined in claim 17.

22. Digital media pre-recorded with a computer program as defined in either Claim 20 or claim 21.

1/1

Figure 1


Key
Calculate embedding Location Ei
File dimensions
Modify at Ei
Data values
Hashing algorithm
1. Signum
2. SHA/MDS
3. User defined
Calculate signatures
Data file
Encryption algorithm
1. Signum
2. RSA
3. User defined
Encrypt S to Se
Embedding algorithm
1. Signum
Embed Se @ Ei
Modify data
Authenticated Data file

INTERNATIONAL SEARCH REPORT

Interm nal Application No
PCT/GB 00/03712

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F1/00 G06T1/00

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F G06T H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data, PAJ

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5 949 885 A (LEIGHTON F THOMSON) 7 September 1999 (1999-09-07) column 2, line 1 -column 3, line 55 column 4, line 42 - line 36 column 5, line 66 -column 6, line 16 column 6, line 54 -column 7, line 15	1-5, 17-22
Y	column 8, line 39 -column 9, line 27	8-11,13, 16
A	---	12
X	US 5 930 369 A (COX INGEMAR J ET AL) 27 July 1999 (1999-07-27) abstract column 6, line 27 -column 7, line 45 column 8, line 61 - line 65 column 9, line 45 -column 11, line 15 --- -/--	1,5,6, 14,15

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
15 December 2000	22/12/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Sigolo, A

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Intern al Application No
PCT/GB 00/03712

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5 930 377 A (POWELL ROBERT, NITZBERG MARK) 27 July 1999 (1999-07-27) column 2, line 38 - line 63 column 4, line 44 - line 64	1,7
Y	US 5 907 619 A (DAVIS DEREK L) 25 May 1999 (1999-05-25) figures 1,2 column 2, line 42 -column 4, line 65	13,16
A		4
Y	WO 97 02522 A (HIGHWATER FBI LTD; HILTON DAVID (GB)) 23 January 1997 (1997-01-23) cited in the application abstract page 2, line 17 -page 3, line 33 page 5, line 29 -page 7, line 35	10,11
Y	WO 98 45768 A (NORTHERN TELECOM LTD) 15 October 1998 (1998-10-15) abstract; figure 3C page 4, line 18 -page 5, line 32 page 11, line 20 -page 15, line 33	8,9

INTERNATIONAL SEARCH REPORT

...ormation on patent family members

Intern ...al Application No
PCT/GB 00/03712

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5949885 A	07-09-1999	US 5664018 A	02-09-1997
		AU 9208298 A	16-03-1999
		EP 1019889 A	19-07-2000
		WO 9910858 A	04-03-1999
		AU 2076097 A	01-10-1997
		EP 0888676 A	07-01-1999
		WO 9734391 A	18-09-1997
US 5930369 A	27-07-1999	AU 701639 B	04-02-1999
		AU 6584096 A	10-04-1997
		CA 2184949 A	29-03-1997
		EP 0766468 A	02-04-1997
		JP 9191394 A	22-07-1997
US 5930377 A	27-07-1999	US 5809160 A	15-09-1998
		US 5721788 A	24-02-1998
		US 6072888 A	06-06-2000
		US 6137892 A	24-10-2000
		CA 2101673 A	01-02-1994
		EP 0581317 A	02-02-1994
		JP 6343128 A	13-12-1994
US 5907619 A	25-05-1999	NONE	
WO 9702522 A	23-01-1997	AU 6233996 A	05-02-1997
		DE 69607844 D	25-05-2000
		EP 0838050 A	29-04-1998
WO 9845768 A	15-10-1998	US 6108420 A	22-08-2000
		AU 6492198 A	30-10-1998
		CN 1255209 T	31-05-2000
		EP 0974084 A	26-01-2000